

## CLAIMS

1. A method for load balancing in a JAVA based environment, the method comprising:

executing an application having a first service module and a control module,  
5 wherein the control module includes application-specific policies for the application;

sensing a utilization of system resources;

generating a second service module using the first service module in response to  
the sensed utilization of system resources;

transferring a state of the first service module to the second service module; and  
10 terminating the first service module.

2. A method as recited in claim 1, wherein the operation of sensing the  
utilization of system resources includes polling system resources.

15 3. A method as recited in claim 1, wherein the operation of sensing the  
utilization of system resources includes receiving notifications from system resources.

4. A method as recited in claim 1, wherein the application-specific policies  
include a specific server on which to generate the second service module.

5        A method as recited in claim 4, wherein the second service module is generated using the specific server.

5        6.        A method as recited in claim 5, wherein the specific server is selected based on the application-specific policies of the control module.

7.        An application having application-specific strategies for use in a JAVA environment, comprising:

10        a plurality of service modules having functionality for the application; and

control module in communication with the plurality of service modules, wherein the control module includes application-specific policies for the application.

8.        An application as recited in claim 7, wherein the control module manages  
15        the service modules.

9.        An application as recited in claim 7, wherein the application-specific policies are programmed using a JAVA programming language.

10. An application as recited in claim 9, wherein the application-specific policies include application-specific load balancing policies.

11. An application as recited in claim 10, wherein a first server module of the plurality of service modules is capable of moving to a second server based on the load balancing policies.

12. An application as recited in claim 11, wherein the control module initiates a generation of a second service module on the second server.

13. An application as recited in claim 12, wherein a state of the first service module is transferred to the second service module.

14. An application as recited in claim 13, wherein the first service module is terminated after the state of the first service module is transferred to the second service module.

15. A method for moving an application within a JAVA environment, comprising the operations of:

executing a first service module and a control module on a first server, the control module having application-specific policies for an application;

5 sending a message from the control module to an executive runtime module, the message requesting the executive runtime module to move the first service module to a second server;

generating a second service module on the second server, the second service module having a state equivalent to a state of the first service module; and

terminating the first service module.

10 16. A method as recited in claim 15, further comprising the operation of obtaining the state of the second service module by a direct transfer from the first service module.

15 17. A method as recited in claim 15, further comprising the operation of obtaining the state of the second service module by using a state server that is shared with the first service module.

18. A method as recited in claim 16, wherein the message from the control module to the executive runtime module includes an identity of the second server.

20

19. A method as recited in claim 15, further comprising the operation of disabling requests to the first service module.

20. A method as recited in claim 19, further comprising the operation of  
5 enabling requests to the second service module.

10